

Final Inteligencia Artificial

Estudiante: Patzi Colodro Sarah Valentina

Carrera: Ingeniería en diseño y animación digital

CU: 104-488

Descripción breve del proyecto y su aplicación real

Descripción del proyecto:
El proyecto consiste en entrenar a un agente utilizando el algoritmo de Q-learning para que aprenda a cruzar una calle de manera segura. En el entorno del proyecto, el agente, representado por un perro, debe navegar un espacio limitado evitando obstáculos móviles (autos) mientras busca alcanzar una meta (la parte superior de la pantalla). Las decisiones del agente se basan en una tabla Q que se actualiza a lo largo de múltiples episodios de entrenamiento.

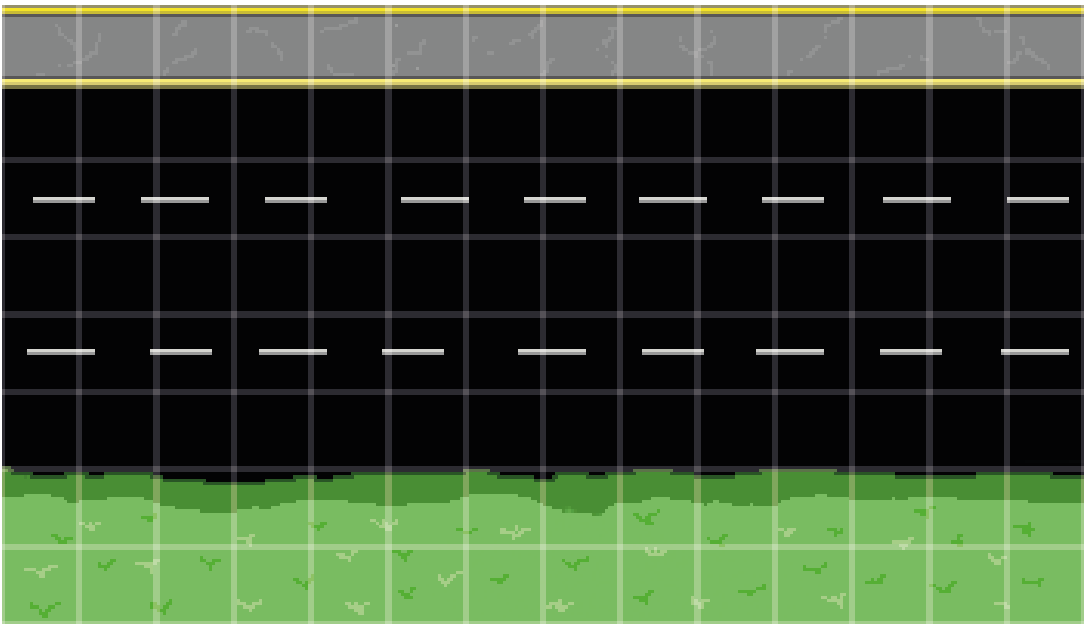
Aplicación en la vida real:
Este modelo puede ser aplicado para diseñar sistemas de asistencia, como perros guía robóticos o algoritmos de navegación inteligentes, enfocados en ayudar a personas ciegas o niños pequeños a cruzar calles de manera segura, evitando obstáculos en entornos urbanos.

Detalles del Proyecto: Cruza la Calle con Q-Learning

Descripción del Entorno

El entorno simula una calle con obstáculos móviles (autos) y un agente (perro) que debe cruzar desde la parte inferior hasta la parte superior de la pantalla. El objetivo del agente es alcanzar la meta de manera segura mientras evita colisiones con los autos en movimiento.

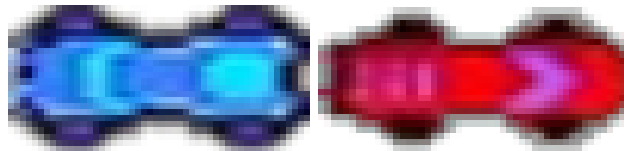
- Dimensiones del entorno:**
La pantalla tiene un tamaño de 350 píxeles de ancho por 200 píxeles de alto. El espacio se divide en una cuadrícula de 14x8 celdas (cada celda de 25x25 píxeles) para representar estados discretos.



- **Agente (Perro):**
Representado por una imagen personalizada (trabajada en pixelart), comienza en la parte inferior central de la pantalla y puede moverse hacia arriba, abajo, izquierda, derecha o permanecer quieto.



- **Obstáculos (Autos):**
Dos automóviles (rojo y azul, trabajados en pixelart) cruzan la pantalla en direcciones opuestas. Sus velocidades y posiciones se reinician al salir del área visible.



Tipo de Escenario

El escenario es **dinámico**, ya que incluye elementos en constante movimiento (los autos) que cambian sus posiciones en cada paso temporal.

Espacio de Acción

El agente dispone de un espacio de acciones discretas compuesto por 5 opciones posibles:

1. **Subir** (hacia la parte superior de la pantalla).
2. **Bajar** (hacia la parte inferior de la pantalla).
3. **Izquierda** (hacia la izquierda).
4. **Derecha** (hacia la derecha).
5. **Quieto** (sin realizar movimiento).

Espacio de Observación / Espacio de Estados

El espacio de observación es discreto y está definido por las posiciones relativas del agente (perro) en una cuadrícula de 14×8 celdas. Cada celda representa un estado único del entorno, derivado de dividir la pantalla de 350×200 píxeles en bloques de 25×25 píxeles.

- **Dimensiones del espacio de observación:** 112 estados (14 columnas x 8 filas).

Recompensas

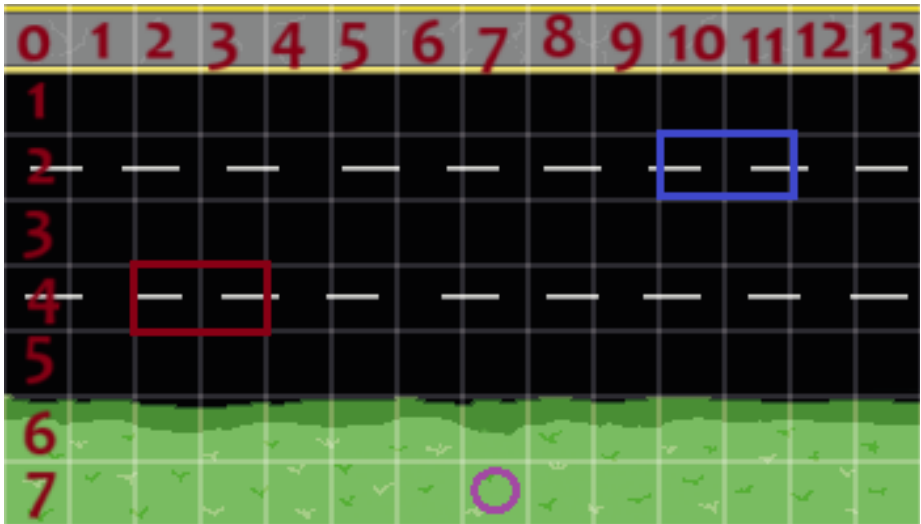
- **+100:** El agente alcanza la meta (parte superior de la pantalla).
- **-1:** Penalización por cada paso para incentivar soluciones eficientes.
- **-100:** Penalización severa si el agente colisiona con un auto.

Estado Inicial

- **Posición del agente:** El agente comienza en el centro de la parte inferior de la pantalla, en la celda correspondiente a la posición [175, 175] en coordenadas pixeladas.
- **Posiciones de los autos:**
 - Auto rojo: [50, 100].
 - Auto azul: [250, 50].

Discretizado:

- **Agente (Perro):** [7,7].
- **Auto rojo:** [2,4].
- **Auto azul:** [10,2]



Estas posiciones se reinician al inicio de cada episodio.

Dinámica de Transición

La dinámica de transición depende de la acción elegida por el agente y las restricciones del entorno:

- El agente se mueve de acuerdo con la acción seleccionada, pero solo si no excede los límites de la pantalla.
- Los autos se desplazan en direcciones opuestas (uno a la derecha y otro a la izquierda), reiniciando su posición al salir de los límites visibles.
- Si ocurre una colisión entre el agente y un auto, el episodio termina inmediatamente con una penalización severa.

Algoritmo de Aprendizaje

El entorno utiliza el algoritmo **Q-Learning** para entrenar al agente.

- **Tasa de aprendizaje (α /alpha):** 0.1
- **Factor de descuento (γ /gamma):** 0.9
- **Estrategia epsilon-greedy:** Inicia con $\epsilon = 1.0$ para alta exploración, disminuyendo exponencialmente hasta un mínimo de 0.1.

Entrenamiento

- **Episodios:** 5000
- **Máximo de pasos por episodio:** 100
- Durante el entrenamiento, se evalúa el desempeño visualizando cada 100 episodios, mostrando el progreso del agente en tiempo real.

Fin del Episodio

Un episodio puede terminar de las siguientes formas:

1. **Éxito:** El agente alcanza la meta (parte superior de la pantalla), obteniendo una recompensa de +100.
2. **Fallo:** El agente colisiona con un auto, recibiendo una penalización de -100.
3. **Límite de pasos:** Si el agente no logra alcanzar la meta dentro de 100 pasos, el episodio termina sin recompensa adicional, penalizando indirectamente la ineficiencia.

Resultados Esperados

Al finalizar el entrenamiento, se espera que el agente aprenda una estrategia óptima para cruzar la calle, minimizando el número de pasos y evitando colisiones. Los resultados se grafican, mostrando la recompensa acumulada y promedios cada 100 episodios.